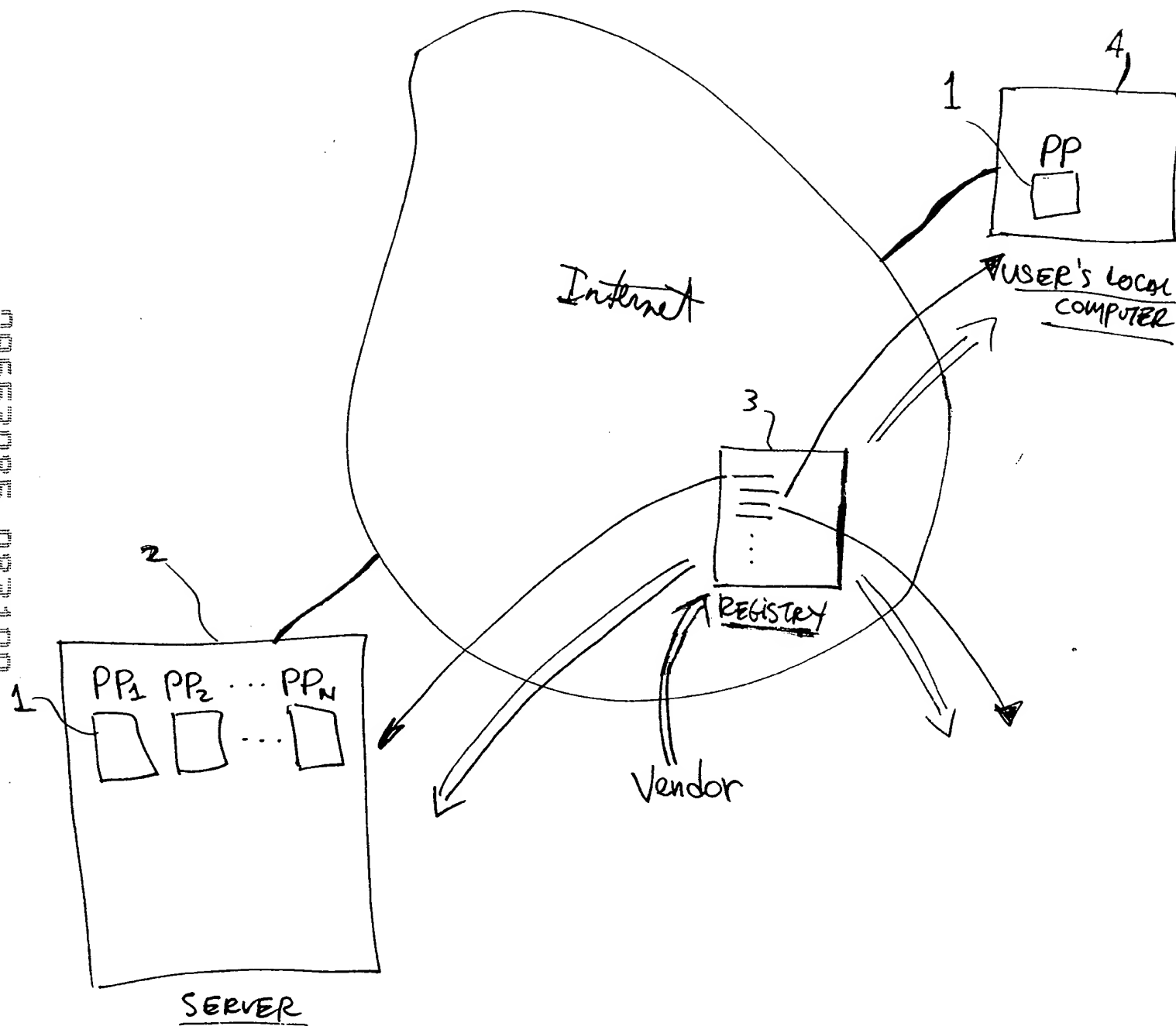


Fig. 1A



→ indicates network address  
⇒ indicates software access

PP = personal page

PP = personal page  
VS = vendor site

S = vendor server  
BR = Bidding Rule

SCAN 111  
Q.

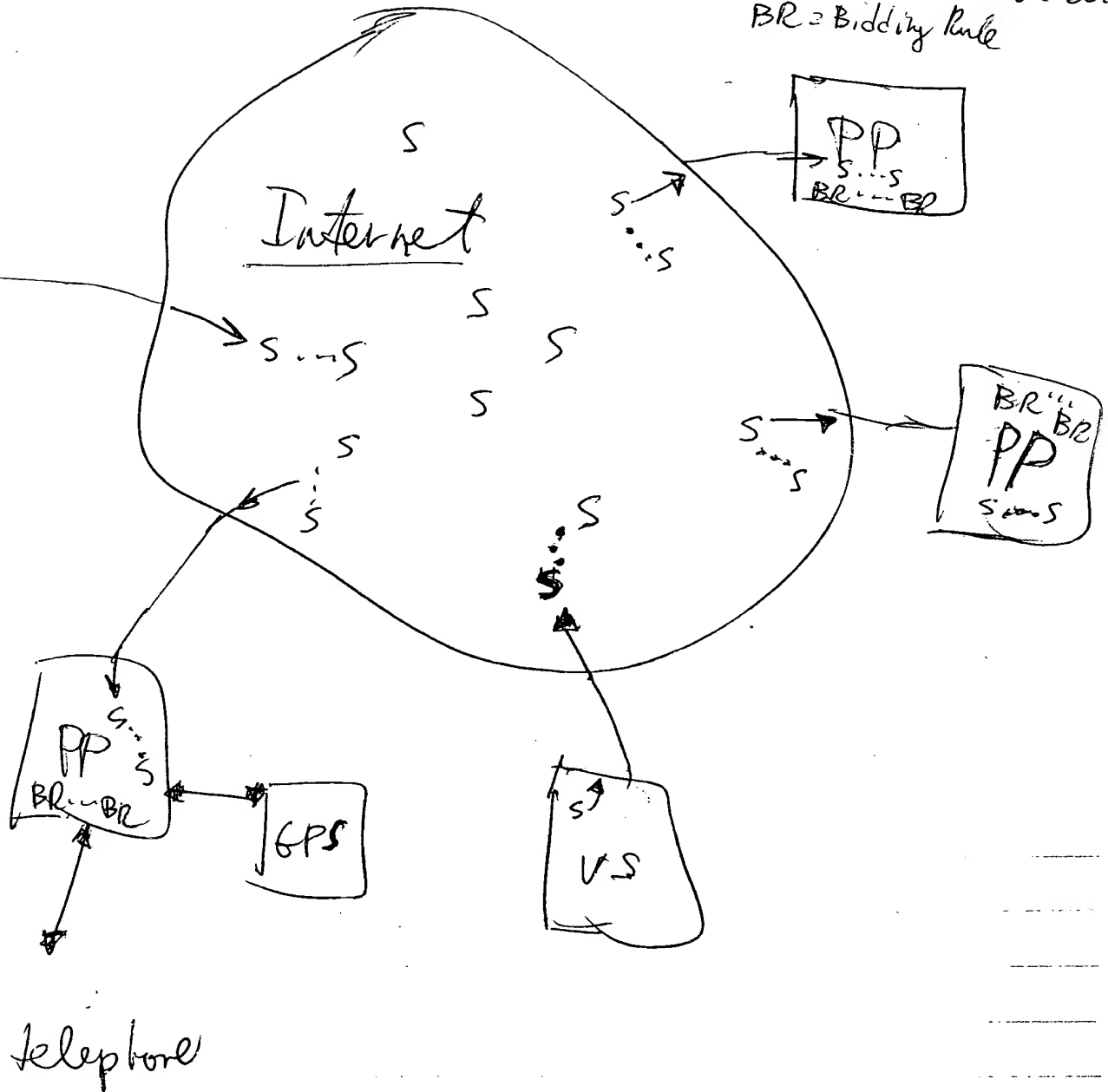


Fig 1B

PRESENT FORM  
TO USER



USER COMPLETES FORM



USER CLICKS  
"ENTER"



COMPLETED FORM  
IS TRANSMITTED TO  
SYSTEM



SYSTEM EXTRACTS  
DATA FROM FIELDS



EXTRACTED DATA  
IS STORED

Fig 2

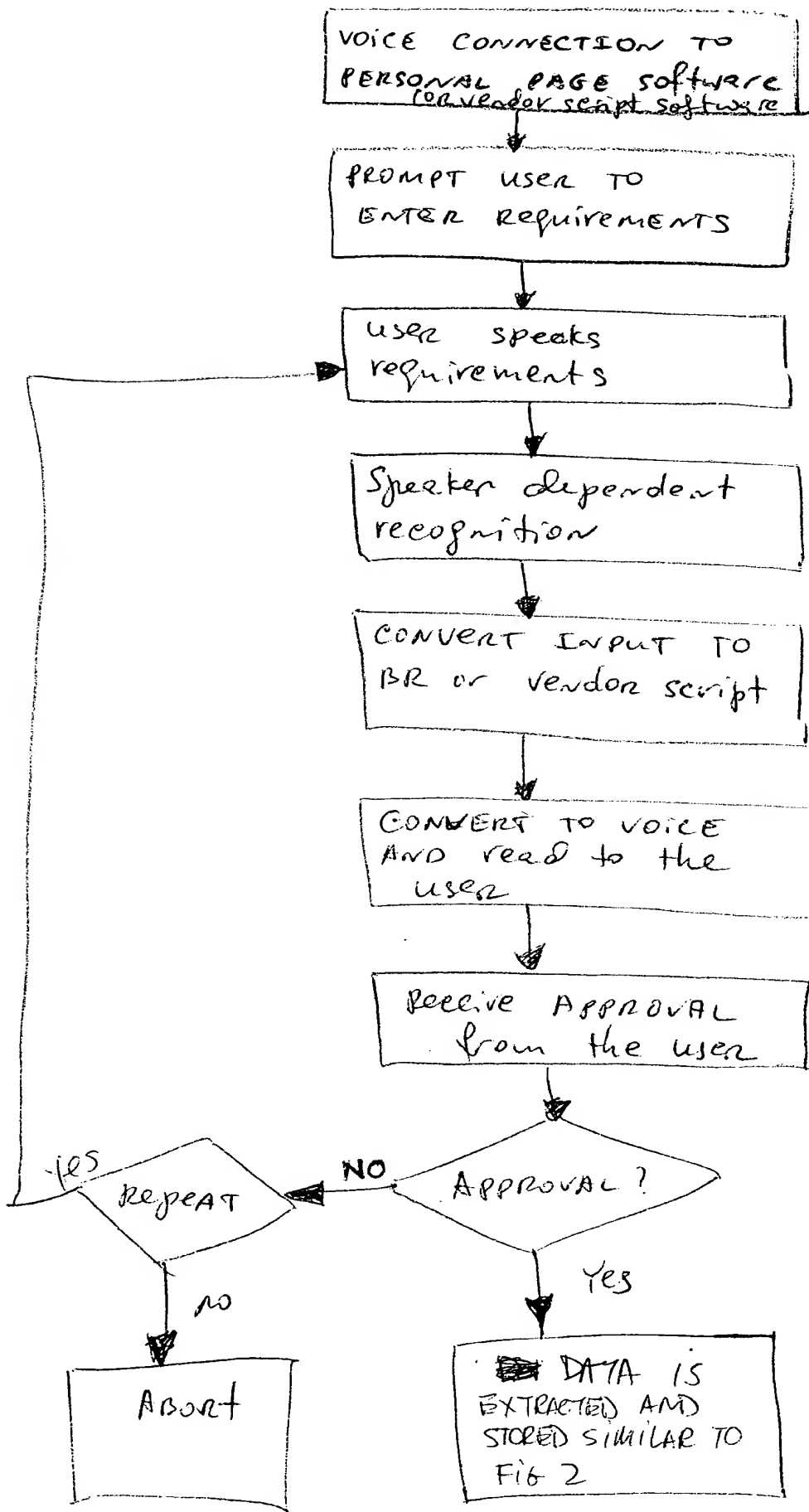


Fig 3

*[Handwritten signature]*

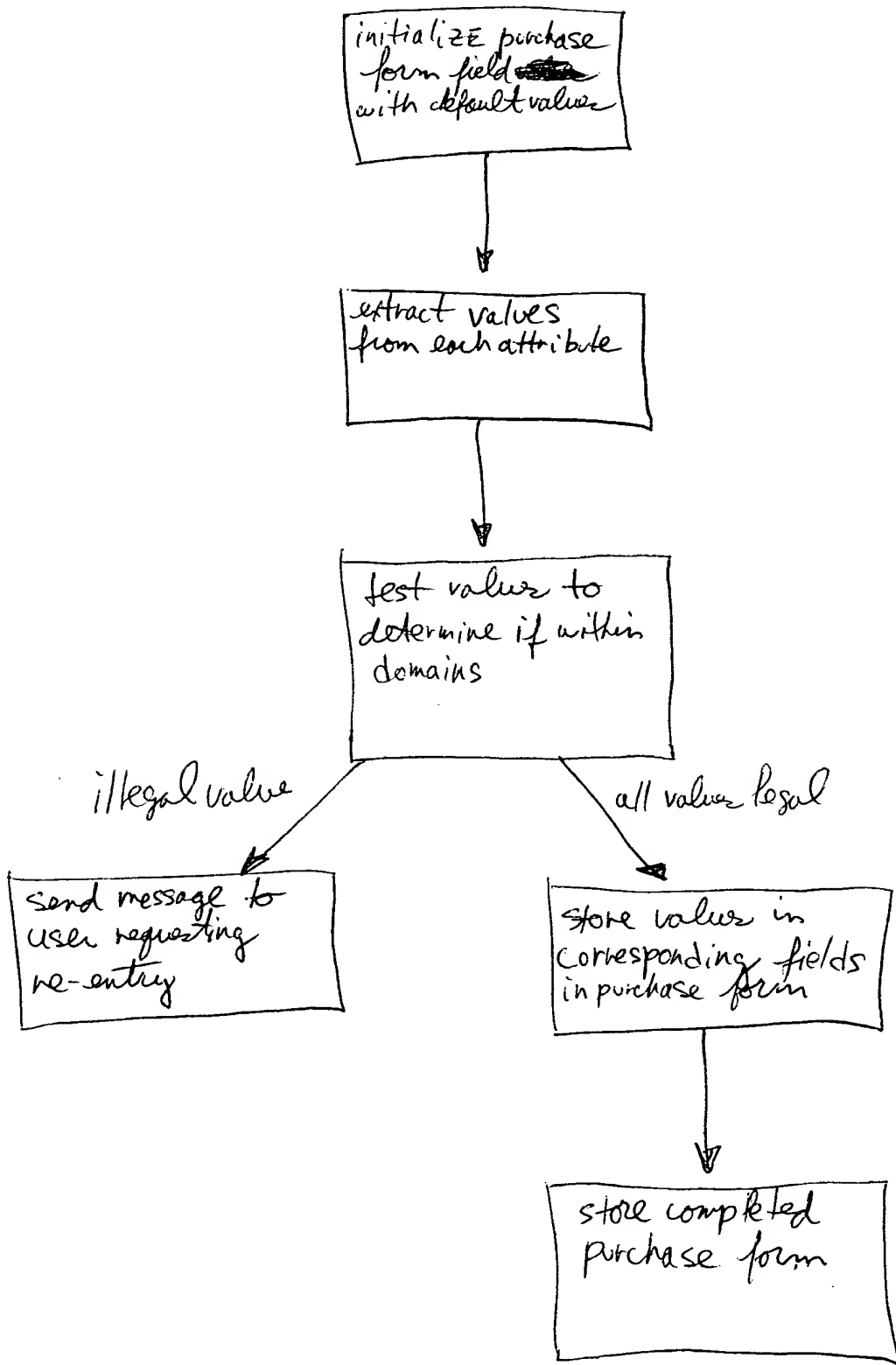


FIG 4

Figs

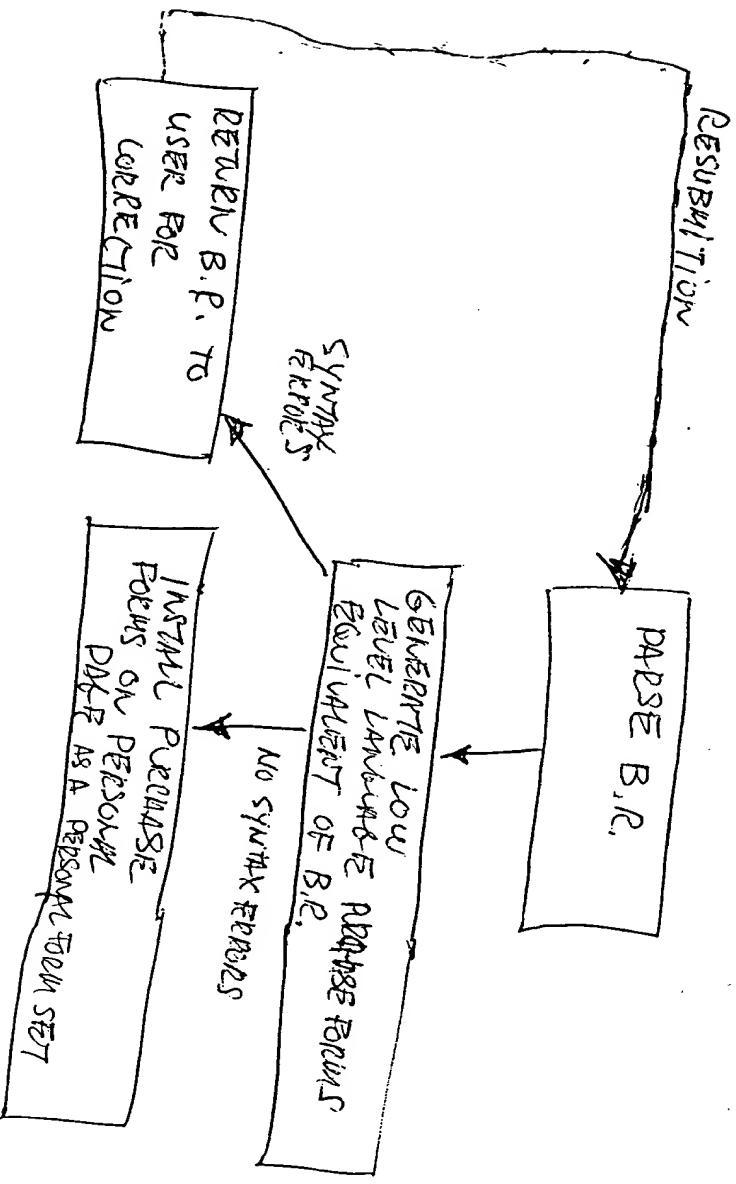
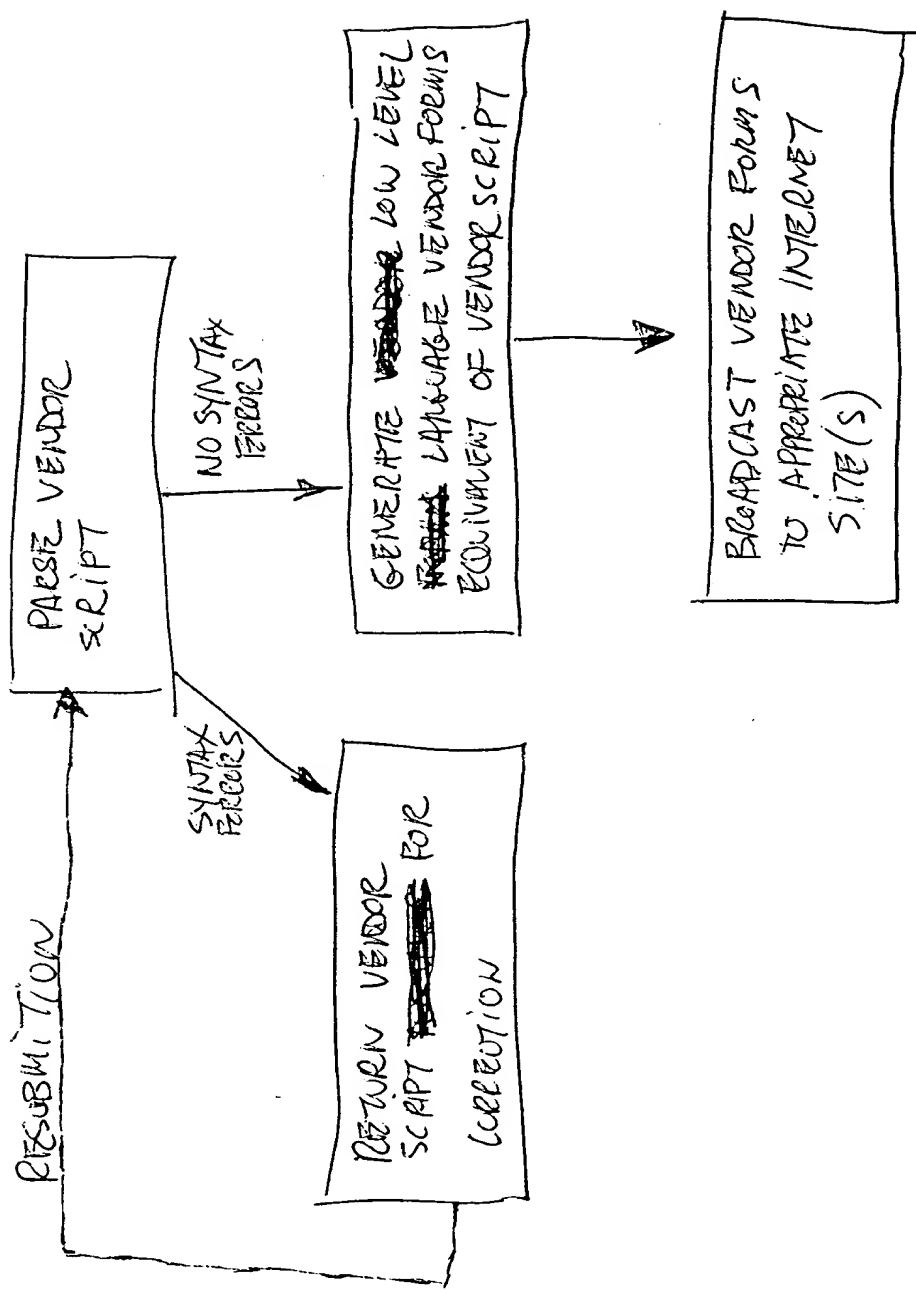


Fig 6



001600 58065300

```

graph TD
    Start([START]) --> Block[Block until arrival of vendor forms]
    Block --> Receive[RECEIVE VENDOR FORMS FROM VENDOR SITE]
    Receive --> Consider{consider all PURCHASE FORM SETS HOSTED HERE}
    Consider -- YES --> Block
    Consider -- NO --> Attempt[ATTEMPT TO SATISFY VENDOR FORMS AGAINST PURCHASE FORM SET]
    Attempt --> Compatible{COMPATIBLE ?}
    Compatible -- YES --> NotifyVendor[NOTIFY VENDOR]
    NotifyVendor --> NotifyUser[NOTIFY USER]
    Compatible -- NO --> Advance[ADVANCE TO NEXT PURCHASE FORM SET]
    Advance --> Consider
    NotifyUser --> Advance
  
```

Fig 7



```

graph TD
    A[EVALUATE BOOLEAN CONDITIONS ASSOCIATED WITH BOTH FORMS] -- BOTH TRUE --> B[CHECK ATTRIBUTES IN FORMS AGAINST EACH OTHER FOR COMPATIBILITY USING DICTIONARY]
    B --> C{ALL ATTRIBUTES COMPATIBLE?}
    C -- NO --> D[RETURN INCOMPATIBLE]
    C -- YES --> E[RETURN COMPATIBLE]
  
```

Fig 8



	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415	2416	2417	2418	2419	2420	2421	2422	2
--	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	---

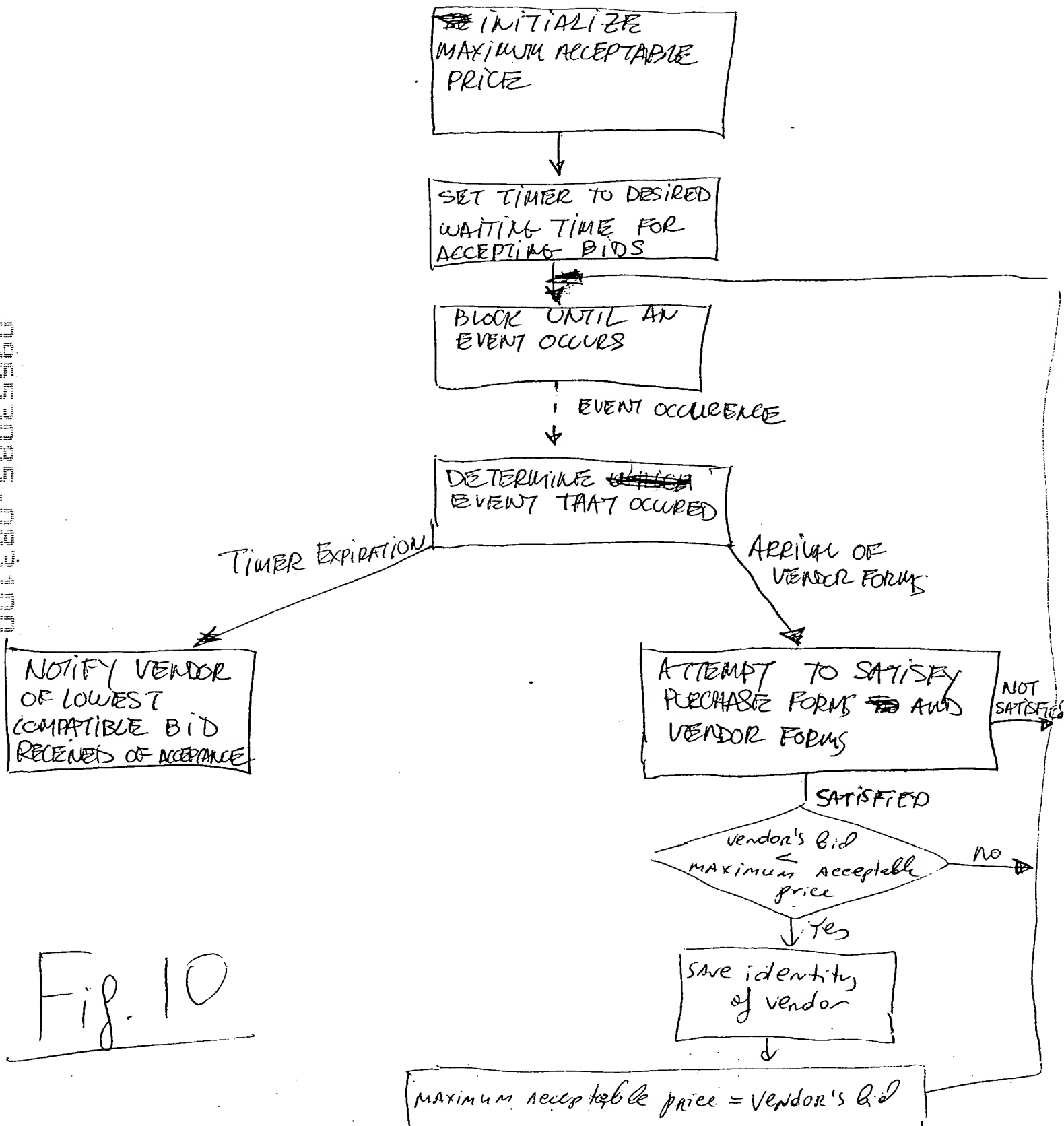


Fig. 10

00553095-09400 007660 50055560

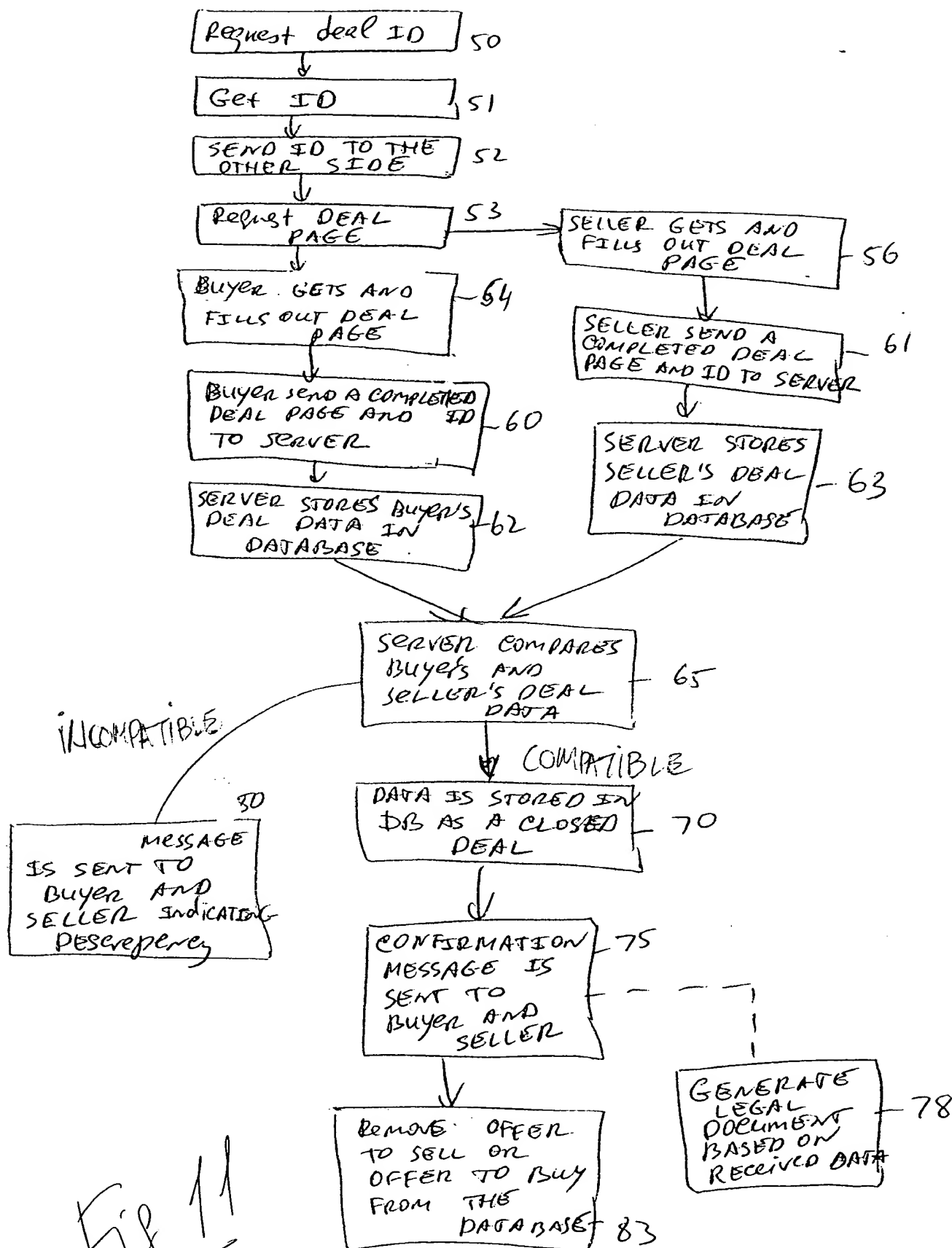


Fig 11

VENDOR BROADCASTS  
ADS TO SITES HOSTING  
PERSONAL PAGES ON  
INTERNET

USER REQUESTS  
ADVERTISEMENT  
REQUEST FORM

SYSTEM SENDS AD  
REQUEST FORM TO  
USER'S LOCAL COMPUTER

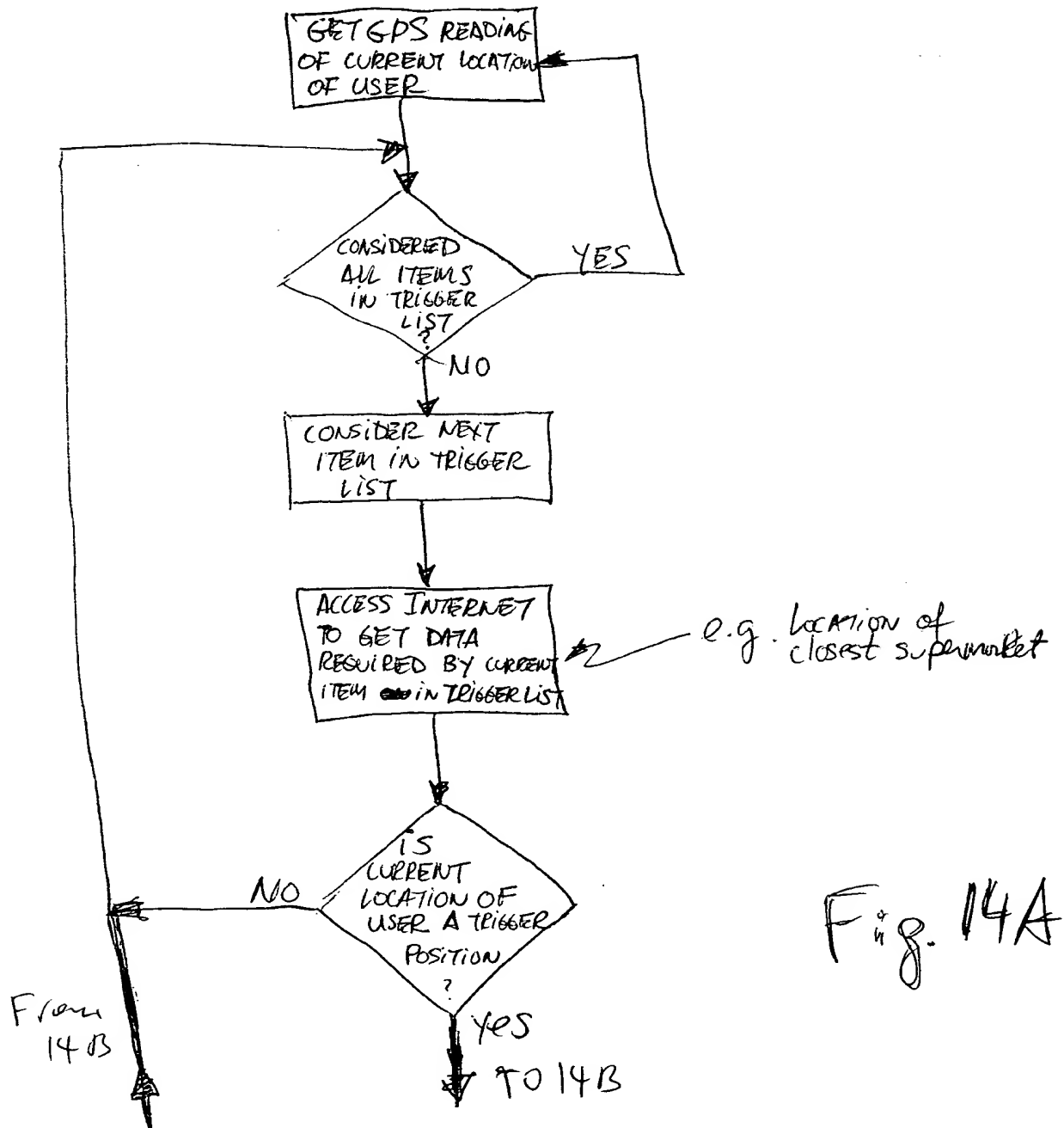
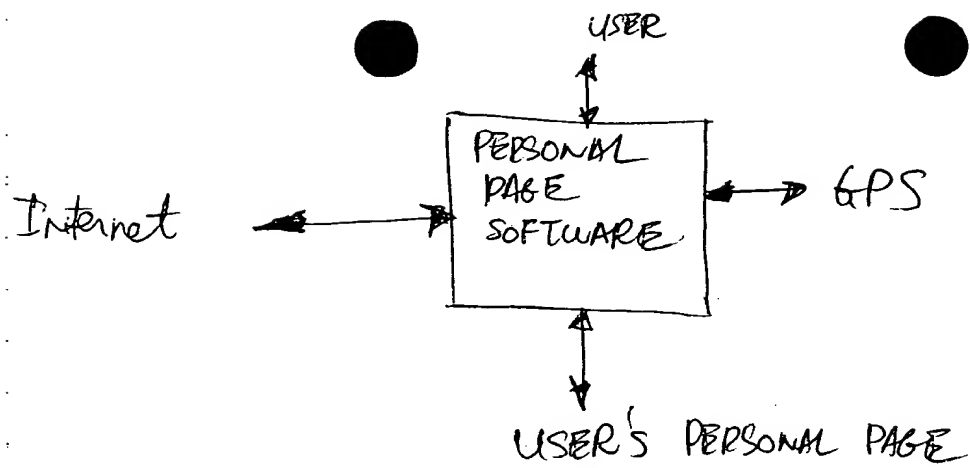
USER COMPLETES AD  
REQUEST FORM AND  
TRANSMITS IT TO SYSTEM

FORM IS PROCESSED AND  
AD REQUEST IS STORED  
ON USER'S PERSONAL PAGE

SYSTEM COMPARES  
VENDOR AD TO  
AD REQUESTS RESIDING  
ON PERSONAL PAGES  
HOSTED AT THIS SITE

ADS ARE SENT  
TO USERS WHOSE  
AD REQUESTS ARE  
COMPATIBLE WITH VENDOR'S AD

Fig 12



6) cont.

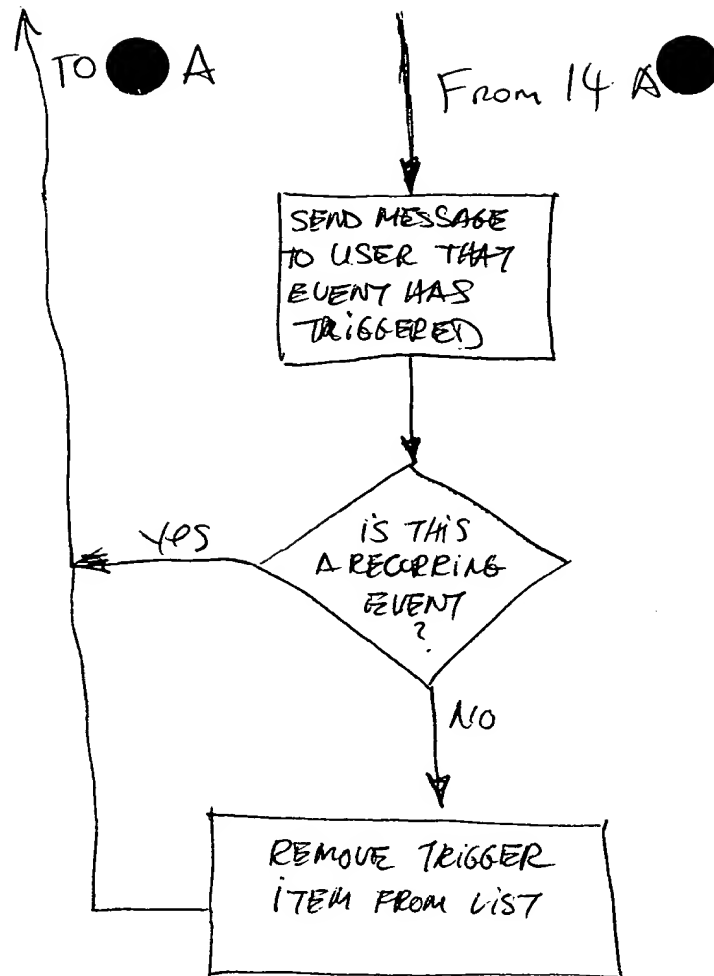


Fig 14 B

00553095-003400

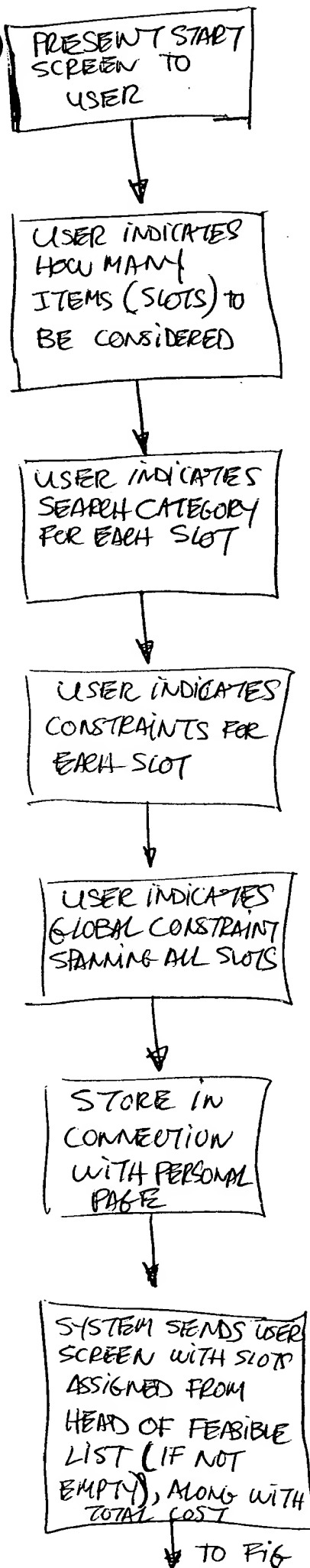


FIG 15A





13

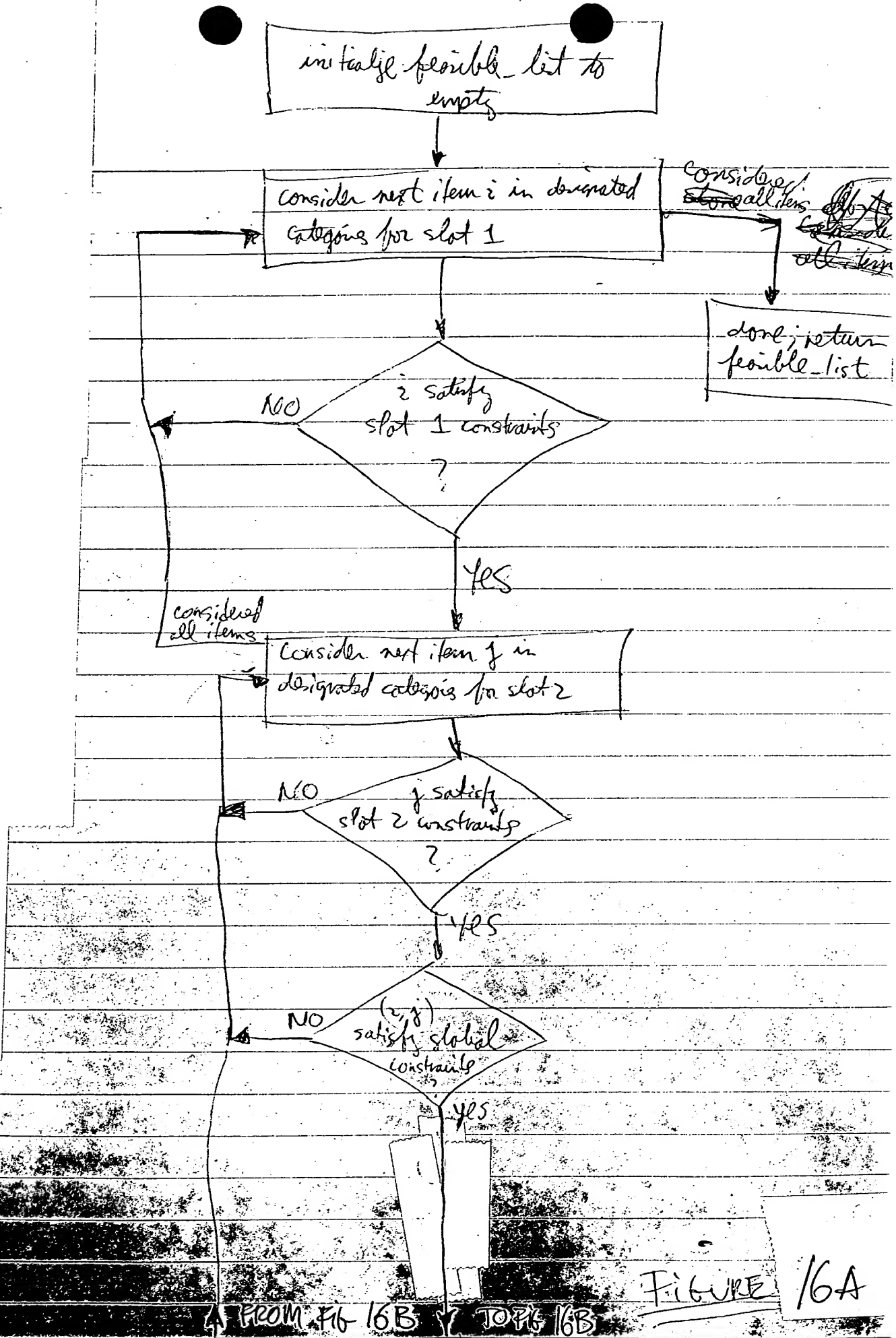


FIGURE 16A

FROM FIG 16B TO FIG 16B

TO FIG. 16A

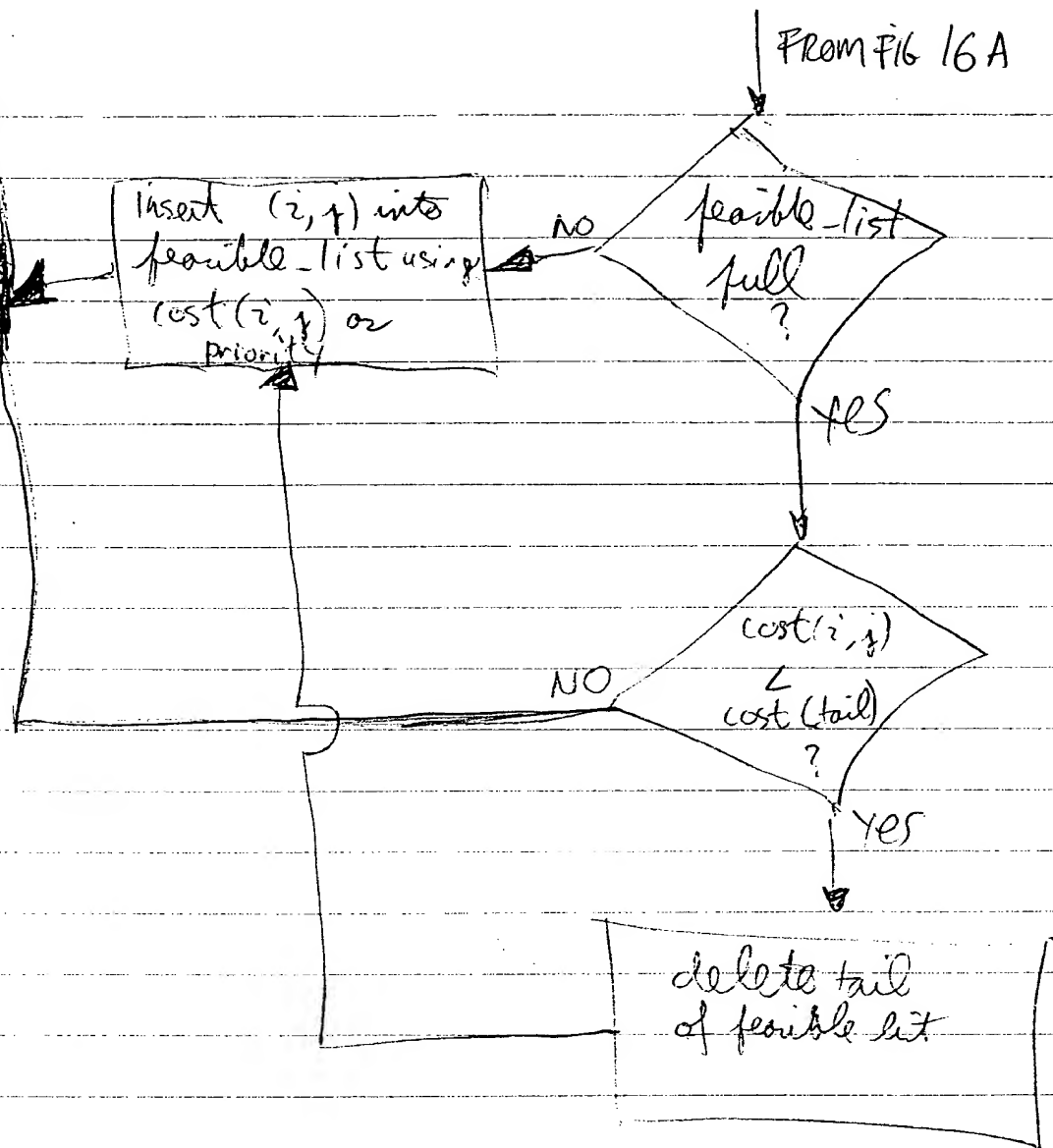


Figure 16B

feasible\_list = empty; // priority queue; ~~cost~~(a,b) is priority

for (i ranging over items in its designated categories)

if (i satisfies slot 1 constraint)

for (j ranging over items in its designated categories)

if (j satisfies slot 2 constraint)

if ((i,j) satisfies global constraints)

if feasible\_list not full

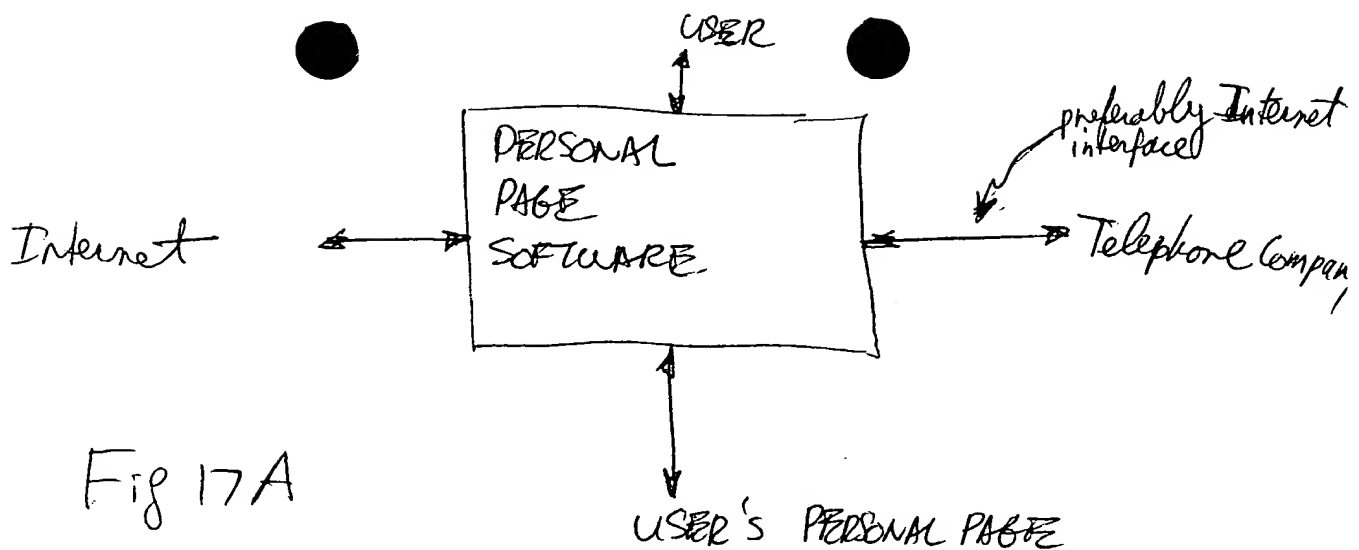
insert (i,j) <sup>and ~~cost~~ cost(i,j)</sup> into feasible\_list;

else if (cost(i,j) < cost(tail))

{ delete tail from feasible\_list;  
insert (i,j) into feasible\_list;  
and ~~cost~~ cost(i,j)  
}

return (feasible\_list);

Fig 16C



USER AT LOCAL COMPUTER  
CREATES NEW OR  
MODIFIES EXISTING  
SERVICE LOGIC  
PROGRAM (SLP)

SERVICE LOGIC PROGRAM  
IS TRANSLATED TO  
LOWER LANGUAGE EQUIVALENT  
LEVEL

TRANSLATED ~~SLP~~  
SLP IS TRANSMITTED  
TO TELEPHONE  
COMPANY

TELEPHONE COMPANY  
INSTALLS ~~SLP~~  
ON TELEPHONE COMPANY  
SWITCHING SYSTEM

TELEPHONE COMPANY  
HANDLES INCOMING CALL  
IN ACCORDANCE WITH  
SLP BY EXECUTING THE SLP

~~PERSONAL PAGE~~  
INCOMING CALL

Fig 17B

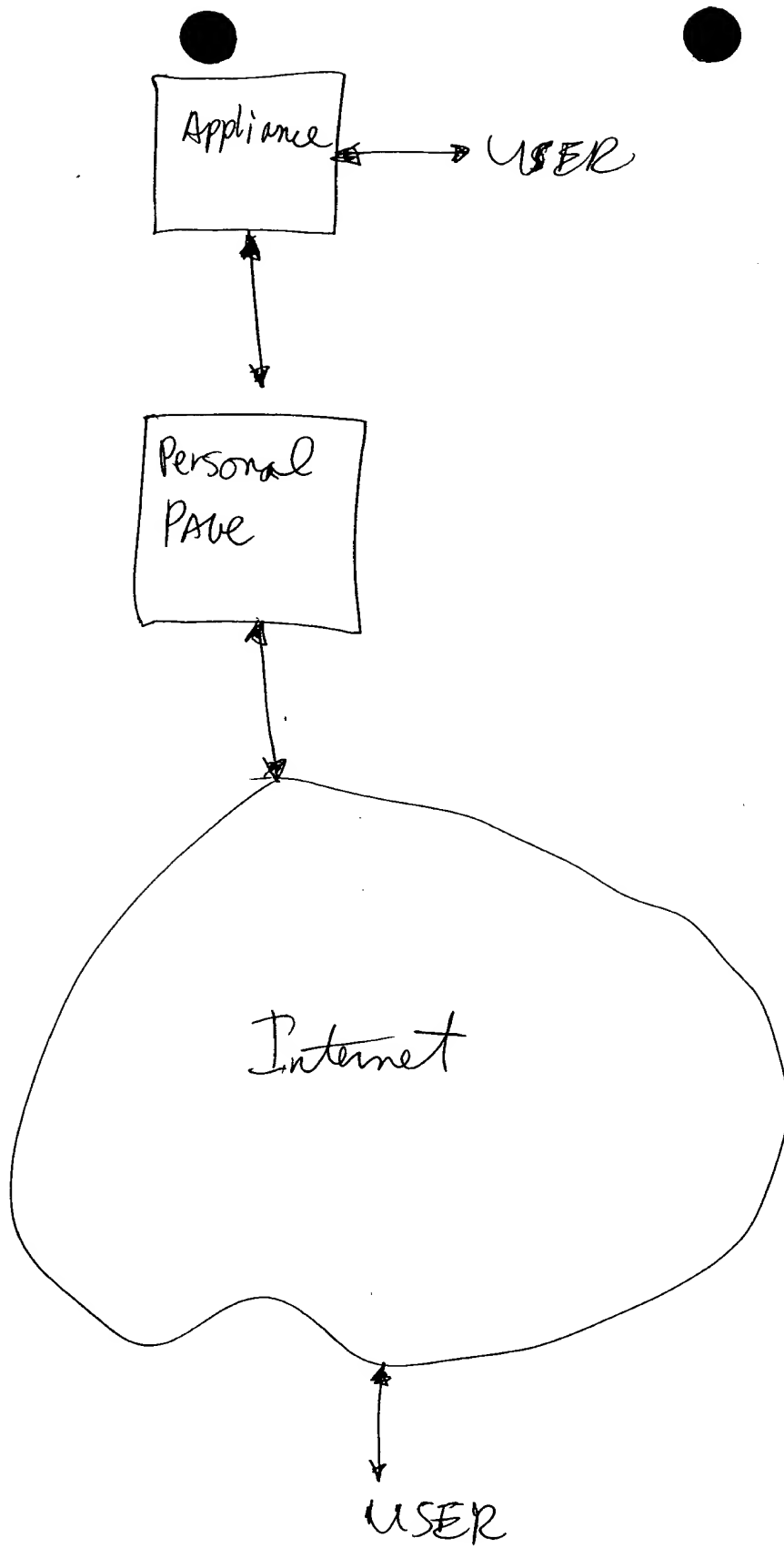


Fig. 18

BR

Vendor Script

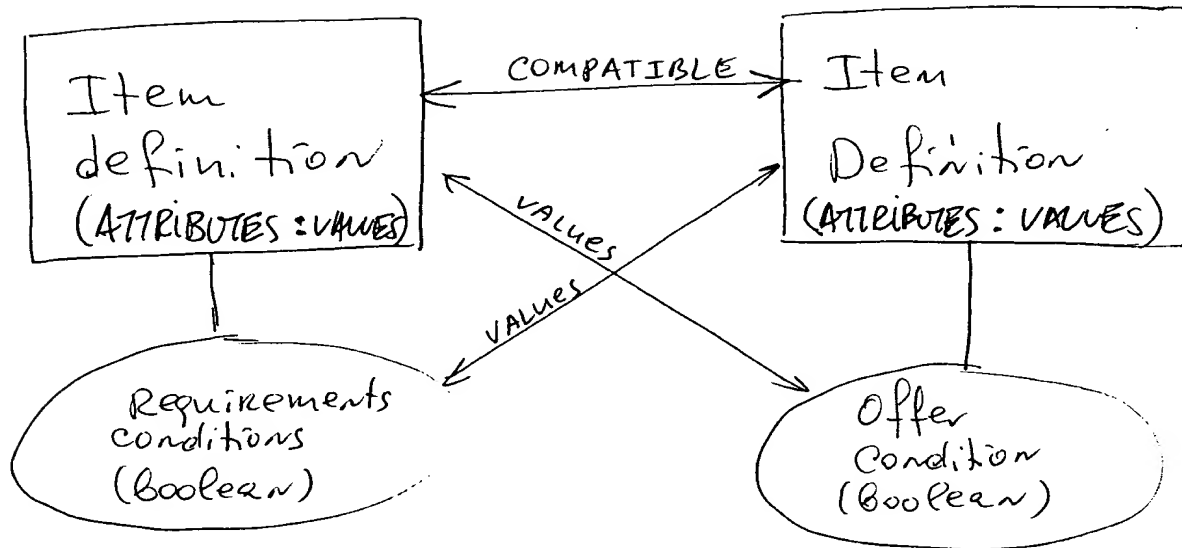


Fig.19

{ ACCEPT\_BR || ACCEPT\_VENDOR\_SCRIPT || MATCHMASTER || SERVICE\_ACTIVATION\_LIST  
|| SERVICE\_SCRIPT\_LIST }

Fig. 20



```

graph TD
    Start(( )) --> Wait[WAIT FOR B.R. MESSAGE FROM USER]
    Wait -- "ACCEPT NEW B.R." --> Insert[INSERT B.R. INTO ACTIVATION LIST AND SIGNAL SERVICE-ACTIVATION LIST PROCESS]
    Wait -- "Replace B.R." --> Replace[REPLACE B.R. IN ACTIVE D.B. OR ACTIVATION LIST]
    Wait -- "DEACTIVATE B.R." --> Delete[DELETE B.R. FROM ACTIVE D.B. OR ACTIVATION LIST]
    Insert --> Send[SEND CONFIRMATION TO USER]
    Replace --> Move[MOVE OLD B.R. TO B.R. HISTORY ARCHIVE]
    Delete --> Move
    Send --> End(( ))
    Move --> End

```

Fig. 21

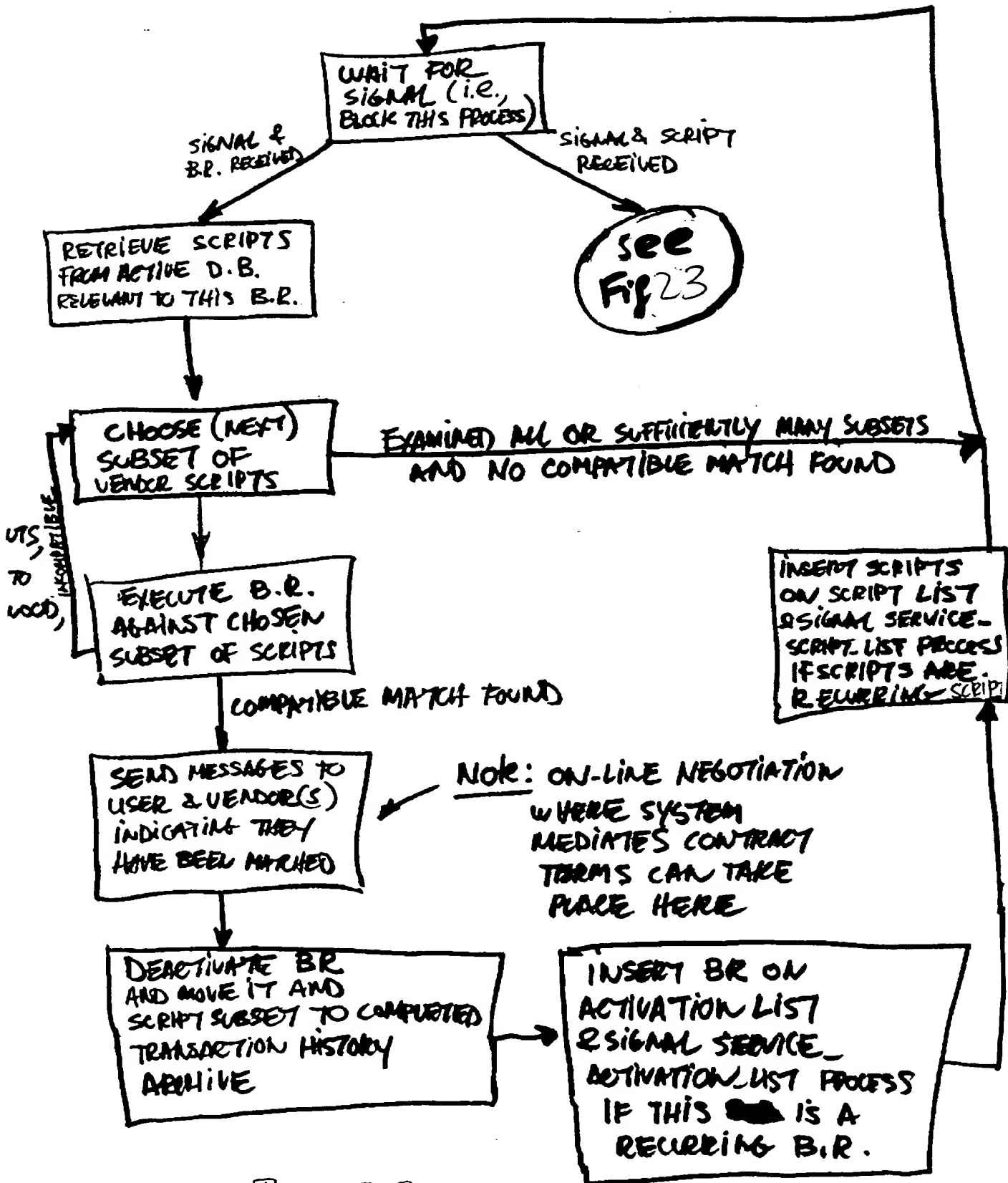


Fig. 22

RETRIEVE B.P.'S  
FROM ACTIVE D.B.  
THAT ARE RELEVANT  
TO THIS SCRIPT

FOR EACH B.P. RETRIEVED,  
INVOKES A NEW, TEMPORARY  
MATCHMAKER PROCESS AS  
SHOWN ON FIG 22, AND  
SEND IT THIS B.P.

Fig 23

001200 50000000

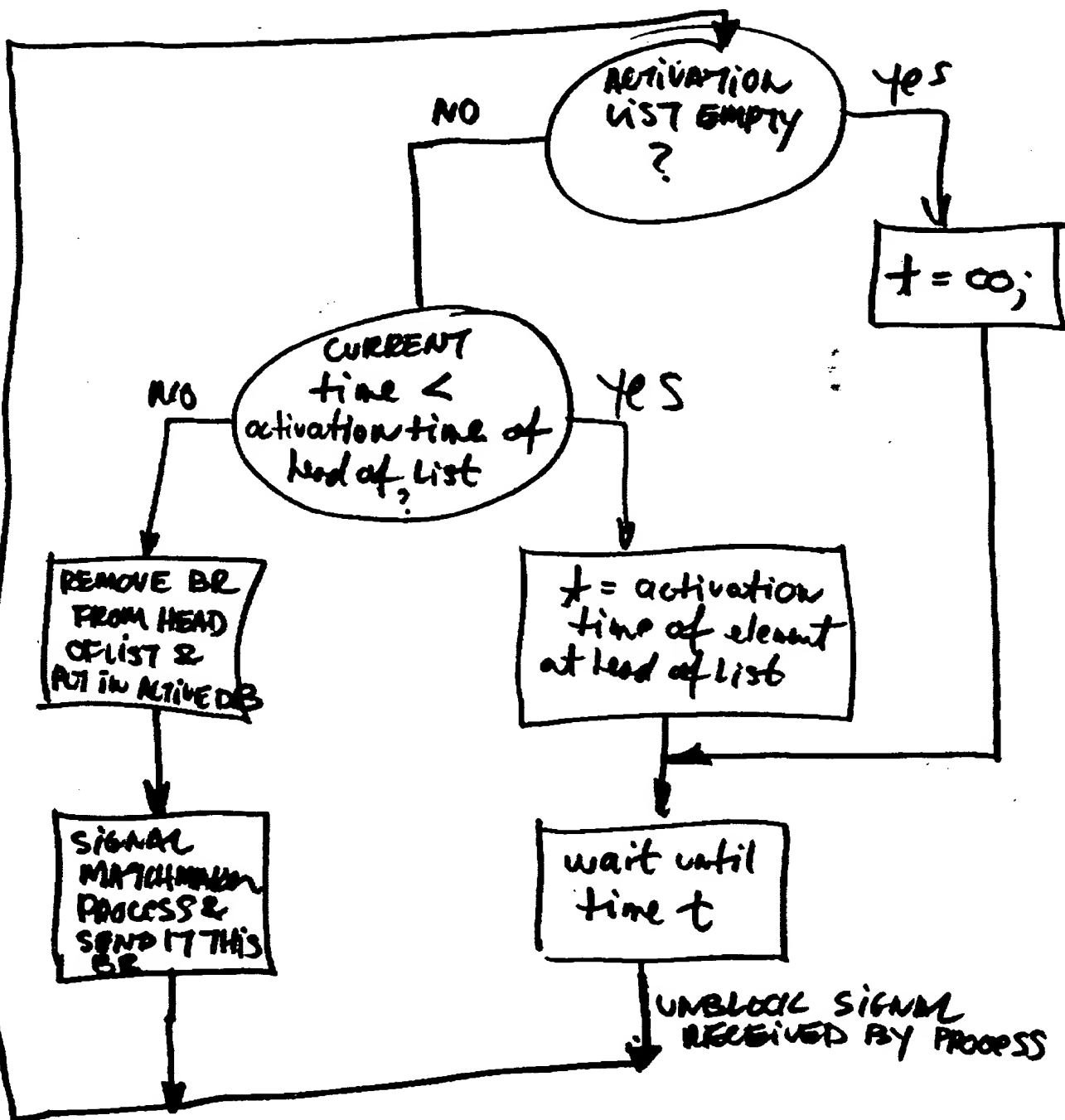


Fig 24